REPORT DOCUMENTATION PAGE	Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
27-02-2015	Final Report	20-Sep-2013 - 31-Oct-2014

4. TITLE AND SUBTITLE
Final Report: Verification and Validation of the Spring Model Parachute Air Delivery System in Subsonic Flow

5a. CONTRACT NUMBER
W911NF-13-1-0291

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
611102

6. AUTHORS
Xiaolin Li

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES
Research Foundation of SUNY at Stony Bro
W-5510 Melville Library

Stony Brook, NY   11794 -3362

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
64478-MA.3

12. DISTRIBUTION AVAILIBILITY STATEMENT
Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT
A mesoscale spring-mass model is used to mimic fabric surface motion.
Through coupling with an incompressible fluid solver, the spring-mass
model is applied to the simulation of the dynamic phenomenon of parachute
inflation. Our model is shown to be numerically convergent under the constraints
that the summation of points masses is constant and
that both the tensile stiffness and the angular stiffness of the

15. SUBJECT TERMS
front tracking, spring model, parachute simulation

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Xiaolin Li
UU	UU	UU	UU		19b. TELEPHONE NUMBER
					631-632-8354

Report Title

Final Report: Verification and Validation of the Spring Model Parachute Air Delivery System in Subsonic Flow

ABSTRACT

A mesoscale spring-mass model is used to mimic fabric surface motion.
Through coupling with an incompressible fluid solver, the spring-mass
model is applied to the simulation of the dynamic phenomenon of parachute
inflation. Our model is shown to be numerically convergent under the constraints
that the summation of points masses is constant and
that both the tensile stiffness and the angular stiffness of the
spring conform with the material's Young modulus and Poisson ratio.
Complex validation simulations conclude the effort via drag force
comparisons with experiments.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

<u>Received</u>   <u>Paper</u>

   TOTAL:

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

<u>Received</u>   <u>Paper</u>

   TOTAL:

Number of Papers published in non peer-reviewed journals:

(c) Presentations

Number of Presentations: 0.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received   Paper

TOTAL:

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received   Paper

TOTAL:

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts):

(d) Manuscripts

Received   Paper

10/01/2014 2.00 Yiyang Yang, Qiangqiang Shi, Xiaolin Li. Application of GPU to Three Computational Models,
Proceedings of IEEE MCSoc Special Session Auto turning of multicore and GPU (06 2014)

10/01/2014 1.00 Qiangqiang Shi, Daniel Reasor, Zheng Gao, Xiaolin Li, Richard Charles. On the Verification and
Validation of a Spring Fabric for Modeling Parachute Inflation,
Journal of Fluid and Structures (06 2014)

TOTAL: **2**

Number of Manuscripts:

Books

Received   Book

  TOTAL:

Received   Book Chapter

  TOTAL:

Patents Submitted

Patents Awarded

Awards

Graduate Students

NAME	PERCENT_SUPPORTED	Discipline
Yiyang Yang	0.70	
Bernard Moore	0.20	
FTE Equivalent:	**0.90**	
Total Number:	**2**	

Names of Post Doctorates

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Names of Faculty Supported

NAME	PERCENT_SUPPORTED	National Academy Member
Xiaolin Li	0.10	
FTE Equivalent:	**0.10**	
Total Number:	**1**	

Names of Under Graduate students supported

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Student Metrics
This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period: 0.00

The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:...... 0.00

The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:...... 0.00

Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):...... 0.00

Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:...... 0.00

The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense 0.00

The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields:...... 0.00

Names of Personnel receiving masters degrees

NAME
Bernard Moore
Total Number: 1

Names of personnel receiving PHDs

NAME
Qiangqiang Shi
Total Number: 1

Names of other research staff

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Sub Contractors (DD882)

Inventions (DD882)

Scientific Progress

See Attachment.

Technology Transfer

Program Director: Joseph Myers
Army Research Office

ARO: W911NF1310291

Verification and Validation of the Spring Model Parachute Air Delivery System

Principal Investigators:
Xiaolin Li
University at Stony Brook

Reporting Period:
September 20, 2013 – October 31, 2014

Recipient:
Research Foundation
University at Stony Brook
Stony Brook, NY 11794-3366

1 ARO Report

1.1 Modelling of Parachute Canopy

Simulation of parachute inflation *via* computational methods has attracted the attention of scientists at the Department of Defense laboratories and academia alike. Some of the successful studies include those reported in [28, 27, 24, 29, 25, 26], where the Deforming-Spatial-Domain/Stabilized Space-Time (DSD/SST) method [34, 35] was used in [27, 24, 29, 25, 26]. The studies in [37, 36, 31, 32, 38] also used the DSD/SST as the core numerical method, but involved new versions and special techniques. These studies successfully addressed the computational challenges in handling the geometric complexities of the parachute canopy and the contact between parachutes in a cluster. Kim and Peskin *et al.* used the immersed boundary method to study the semi-opened parachute in both two and three dimensions [10, 12], their simulations are on small Reynolds number (about 300) and applied to parachute with payload of several grams. Karagiozis used the large-eddie simulation to study the parachute in Mach 2 supersonic flow [7]. Purvis [20, 21] used springs to represent the structures of the fore-body, the suspension lines, and the canopy, etc. In these papers, the authors used cylindrical coordinates with the center line as the axis. In the paper by Strickland *et al.*[30], the authors developed an algorithm called PURL to couple the structure dynamics (PRESTO) and fluid mechanics (CURL), in which mass is added to each of the structure node based on the diagonally added mass matrix and a pseudo is computed from the fluid code which is the sum of the actual pressure and the pressure associated with the diagonally added mass. Tutt and Taylor [41, 40] simulated the parachute through the LS-DYNA code. They used an Eulerian-Lagrangian penalty coupling algorithm and multi-material ALE capabilities with LS-DYNA to replicate the inflation of small round canopies in a water tunnel.

Our computational approach to the simulation of the parachute delivery system is based on the front tracking platform. In the last ARO funded project during which the PI served as the principal developer, we established a computational platform for the parachute study by employing the spring model for the parachute canopy and the string chords. We designed a set of new data structure to allow the application of the *FronTier* library to track the dynamic motion of fabric surface driven by the gravitational force of payload and the fluid pressure. We discretize the fabric surface into a homogeneously triangulated surface mesh.

1.1.1 Analysis of the Spring System

We made a detailed analysis of the spring system which is used to model the fabric surface of the parachute canopy (Figure 1). When no external driving force is applied, the fabric surface, which is represented by the spring-mass system, is a conservative system whose total energy (kinetic energy plus potential energy) is a constant. Assuming each mesh point represents a point mass m in the spring system with position vector $\mathbf{x}_i$, the kinetic energy of the mass point i is $T_i = \frac{1}{2}m|\dot{\mathbf{x}}_i|^2$, where $\dot{\mathbf{x}}_i$ is the time derivative, or velocity vector of the mass point i. Here we report the work under the support of ARO grant W911NF1310291 on the improvement of the spring model, its verification of Young's modulus and Poisson ratio,
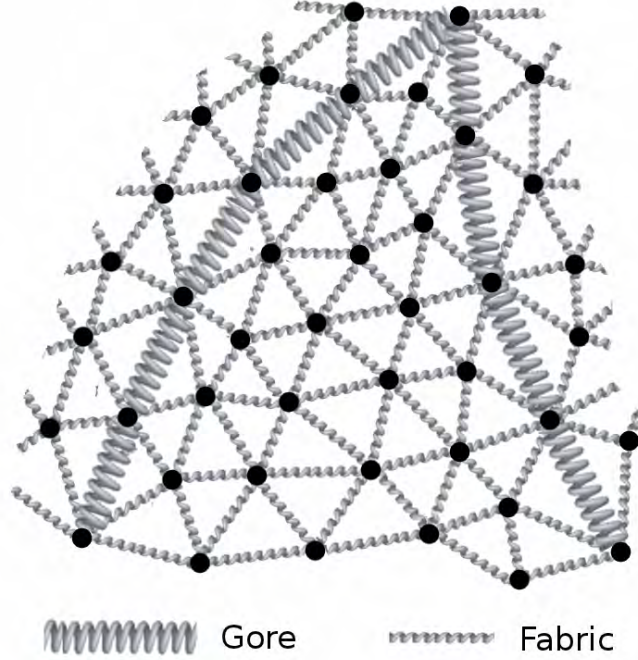
Figure 1: The spring model on a triangulated mesh. Each vertex point in the mesh represents a mass point with point mass m. Each edge of the triangles has an equilibrium length set during initialization and the changing length exerts a spring force on the two neighboring vertices in opposite directions. Gores are added as curves with larger spring constant. This plot shows the mesh of canopy with gores.

and the validation in the simulation of parachute inflation and descending.

The original spring model [17] that serves as the basis for this effort is a simplified mesoscale model which assumes the force required to bend the surface is negligible and the force to stretch the surface is proportional to the displacement from the equilibrium distance between adjacent mass points. In this model, the kinetic energy and the potential energy of the triangulated mesh is given by

$$T_i = \frac{1}{2} m_i |\dot{\mathbf{X}}_i|^2, \quad V = \frac{1}{4} \sum_{i=1}^{N} \sum_{j=1}^{N} k(|\mathbf{X}_i - \mathbf{X}_j| - l_{ij}^0)^2 \eta_{ij}, \tag{1}$$

where $\mathbf{X}_i$ is the position of the vertex i, k is the spring constant, l_{ij}^0 is the equilibrium length of the side shared by vertices $\mathbf{X}_i$ and $\mathbf{X}_j$, and η_{ij} is a boolean variable for adjacency. Delingette modification is added to the spring model. This has made possible for a direct link between the meso-scale spring model and the elastic membrane model in continuum mechanics.

Illustrative features of Delingette's model are given in Figure 2. The energy of membrane $W(T_{\mathbf{X}_0})$ that is required to deform a single triangle $T_{\mathbf{X}_0}$ with vertices $\{\mathbf{X}_{10}, \mathbf{X}_{20}, \mathbf{X}_{30}\}$ into

its deformed position $T_{\mathbf{X}}$ with vertices $\{\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3\}$ consists of two parts,

- The energy of three tensile springs that prevent edges from stretching.

- The energy of three angular springs that prevent any change of vertex angles.

For a triangle in equilibrium $T_{\mathbf{X}_0}$, the initial states are given by area $A_{\mathbf{X}_0}$, angles α_i, and lengths l_i^0 ($i \in 1, 2, 3$) in equilibrium while $A_{\mathbf{X}}$, β_i and l_i denote the area, angles, and lengths of the deformed triangle $T_{\mathbf{X}}$, respectively.

The edge elongation can be written as $dl_i = l_i - l_i^0$. The potential energy is given [2] as

$$W(T_{\mathbf{X}_0}) = \sum_{i=1}^{3} \frac{1}{2} k_i^{T_{\mathbf{X}_0}} (dl_i)^2 + \sum_{\substack{i=1 \\ j=(i+1) \pmod 3 \\ k=(i+2) \pmod 3}}^{3} \gamma_i^{T_{\mathbf{X}_0}} dl_j dl_k$$

where

$$k_i^{T_{\mathbf{X}_0}} = \frac{(l_i^0)^2 (2 \cot^2 \alpha_i (\lambda + \mu) + \mu)}{8 A_{\mathbf{X}_0}}$$

is the tensile stiffness and

$$\gamma_i^{T_{\mathbf{X}_0}} = \frac{l_j^0 l_k^0 (2 \cot \alpha_j \cot \alpha_k (\lambda + \mu) - \mu)}{8 A_{\mathbf{X}_0}} \tag{2}$$

is the angular stiffness, where $j = (i+1) \pmod 3$ and $k = (i+2) \pmod 3$. γ and μ are the Lamé coefficients of the material. These coefficients are simply related to the two physically meaningful parameters defined in planar elasticity for a membrane, that is, Young's modulus E and the Poisson ratio ν [3]:

$$\lambda = \frac{E\nu}{1 - \nu^2} \quad \text{and} \quad \mu = \frac{E(1 - \nu)}{1 - \nu^2}.$$

Young's modulus quantifies the stiffness of the material, whereas the Poisson ratio characterizes the material compressibility.

Through the application of Rayleigh-Ritz analysis the fabric surface, represented by the triangular mesh, should evolve by minimizing its membrane energy. Therefore, along the opposite derivative of that energy with respect to the nodes of the system, that is, the deformed positions $\mathbf{X}_i$:

$$\begin{aligned} \mathbf{F}_i(T_{\mathbf{X}_0}) &= -\frac{\partial W(T_{\mathbf{X}_0})}{\partial \mathbf{X}_i} \\ &= \sum_{j \neq i} k_j^{T_{\mathbf{X}_0}} (dl_j) \frac{\mathbf{X}_k - \mathbf{X}_i}{l_j} + \sum_{j \neq i} (\gamma_k^{T_{\mathbf{X}_0}} dl_i + \gamma_i^{T_{\mathbf{X}_0}} dl_k) \frac{\mathbf{X}_k - \mathbf{X}_i}{l_j} \end{aligned} \tag{3}$$

The membrane deformation energy of the whole triangulation is the sum of the energy of each triangle. Thus, we obtain the force at each vertex point as Eq. (4).
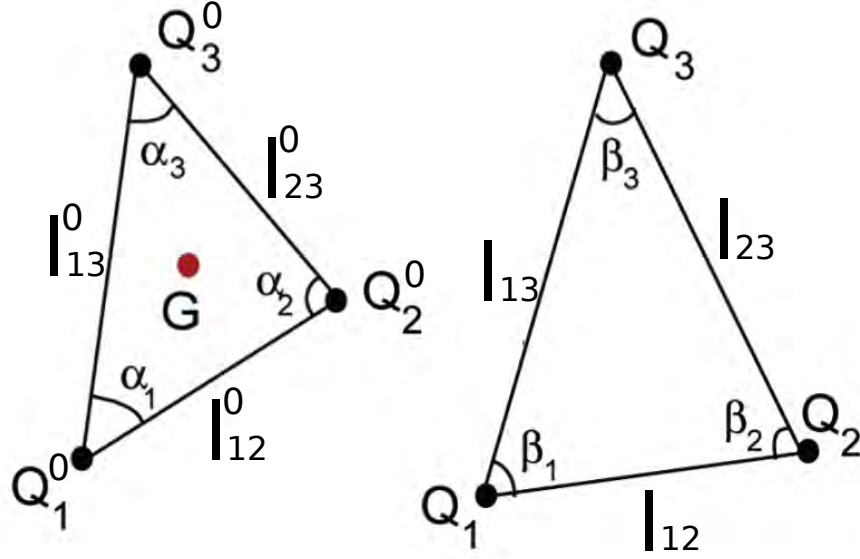
3

Figure 2: (a) Rest triangle $T_{\mathbf{X}_0}$ whose vertices are $\mathbf{X}_{i0}$. (b) Deformed triangle $T_{\mathbf{X}}$ whose vertices are $\mathbf{X}_i$.
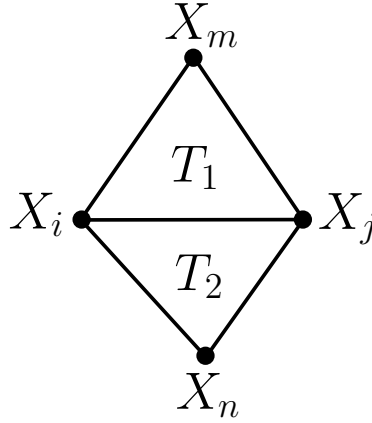

Figure 3: Triangles T_1 and T_2 share $\mathbf{X}_i$ and $\mathbf{X}_j$, the other vertices of triangles T_1 and T_2 are $\mathbf{X}_m$ and $\mathbf{X}_n$ respectively.

$$\mathbf{F}_i = \sum_{j=1}^{N} \eta_{ij} \mathbf{F}_{ij} \tag{4}$$

As illustrated by Figure 3, if $\mathbf{X}_i$ and $\mathbf{X}_j$ are shared by two triangles T_1 and T_2, and the

4

other vertices of triangles T_1 and T_2 as denoted as $\mathbf{X}_m$ and $\mathbf{X}_n$ respectively, we have

$$\begin{aligned}
\mathbf{F}_{ij} &= ((k_{ij}^{T_1} + k_{ij}^{T_2})dl_{ij} + (\gamma_i^{T_1}dl_{im} + \gamma_j^{T_1}dl_{jm} + \gamma_i^{T_2}dl_{in} + \gamma_j^{T_2}dl_{jn}))\mathbf{e}_{ij} \\
&= \tilde{k}_{ij}dl_{ij}\mathbf{e}_{ij} + \tilde{\gamma}_{ij}dl_{ij}\mathbf{e}_{ij} \tag{5}
\end{aligned}$$

where $\tilde{k}_{ij} = k_{ij}^{T_1} + k_{ij}^{T_2}$, $\tilde{\gamma}_{ij} = (\gamma_i^{T_1}dl_{im} + \gamma_j^{T_1}dl_{jm} + \gamma_i^{T_2}dl_{in} + \gamma_j^{T_2}dl_{jn})/dl_{ij}$ and $\mathbf{e}_{ij}$ is the unit vector from $\mathbf{X}_i$ to $\mathbf{X}_j$.

If the second term in Eq. (5) is neglected, Delingette's model is the same as model used in [9] except that the spring constant varies if the corresponding initial triangles deviate from isosceles triangles. Numerical evidence suggests that both the variation of $\tilde{k}_{ij}$ and the modification from the second term (due to angular stiffness) are significant in the above cases.
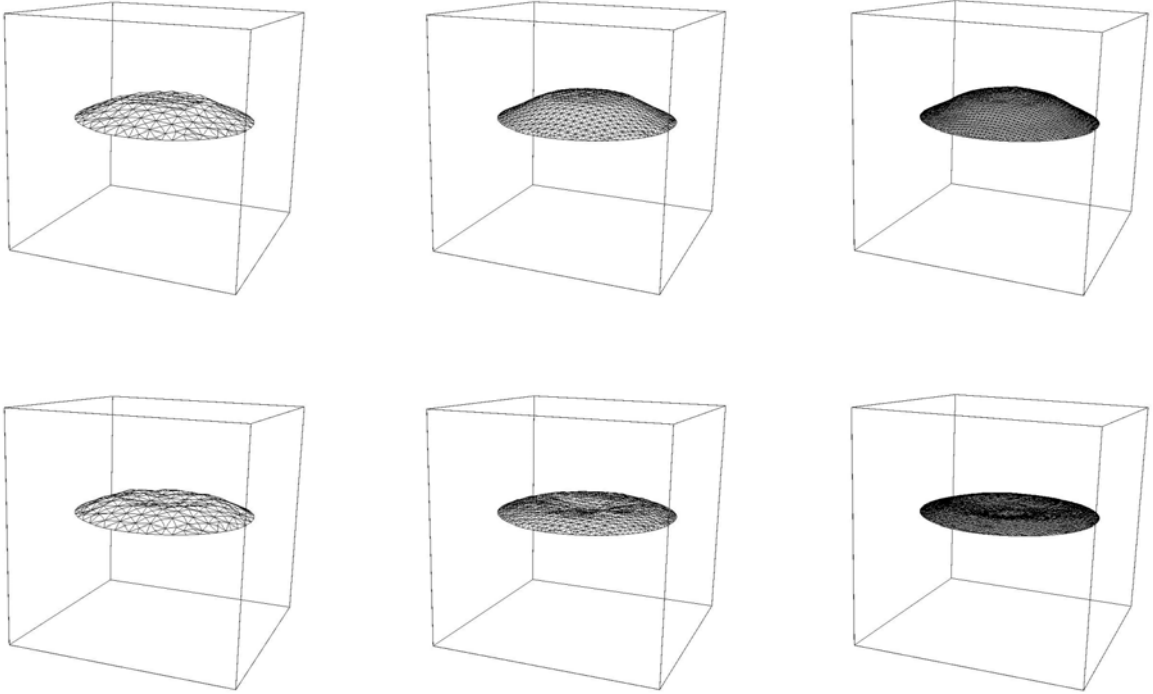

Figure 4: Convergence test of the drum membrane under mesh refinement. From left to right the computational mesh of the domain are 15, 30, and 60 respectively. The total mass of the membrane is kept constant in the simulations. The upper three plots show the membrane position at $t = 1$ and the lower plots show the membrane position at $t = 2$.

1.1.2 Numerical Convergence Verification

Numerical convergence under computational mesh refinement is a crucial step in assessing the validity of spring-mass model. For this purpose, we carried out two sequences of simu-

reference mesh size	e_A	e_k	e_p
15 and 30	0.02528	1.91810	1.97967
30 and 60	0.01507	1.21784	1.21772
60 and 120	0.00604	0.53550	0.52837

Table 1: Convergence tests of spring model for a fabric drum. In the computational sequences, the total mass of the membrane is fixed. As the number of points increases, the point mass is reduced accordingly. Cauchy error is calculated on two consecutive mesh sequences. Column e_A, e_k, and e_p are errors of total area, total kinetic energy and total spring potential energy respectively. The numerical results show the first order convergence.

lations with increasing number of grid points. Our convergence study includes both a one dimensional string in the two dimensional plane and a two dimensional membrane in the three dimensional space.

In one case of the verification study, a circular vibrating membrane with radius of $0.4m$ and total weight of $380g$ is considered. The membrane is linearly perturbed at initial time and its circular boundary is fixed. A sequence of four cases with computational mesh 15^3, 30^3, 60^3, and 120^3 are computed. The errors of total area are shown in Table 1. From Table 1, we can clearly see that the Cauchy errors are decreasing as the computational mesh is refined. The convergence rate for the membrane is also of first order.

1.1.3 Verification of Young's Modulus and Poisson's Ratio

Young's modulus, also known as the tensile modulus or elastic modulus, is a measure of the stiffness of an elastic material and is a parameter used to characterize materials. It is defined as the ratio of the stress along an axis over the strain along that axis in the range of stress in which Hooke's law is valid. In solid mechanics, the slope of the stress-strain curve at any point is called the tangent modulus. The tangent modulus of the initial, linear portion of a stress-strain curve is called Young's modulus. Young's modulus, E, can be calculated by dividing the tensile stress by the tensile strain in the elastic portion of the stress-strain curve:

$$E = \frac{\text{tensile stress}}{\text{tensile strain}} = \frac{\sigma}{\epsilon} = \frac{F/A_0}{\Delta l/l_0} = \frac{Fl_0}{A_0\Delta l} \tag{6}$$

where E is Young's modulus, F is the force exerted on the object under tension, A_0 is the original cross-sectional area through which the force is applied, Δl is the change of length of the object from the original length l_0 of the object.

Poisson's Ratio is the negative ratio of transverse strain to axial strain. When a material is stretched, it usually tends to contract in the directions transverse to the direction of stretching. The Poisson ratio is the ratio of relative contraction to relative stretching.

$$\nu = -\frac{d\epsilon_{\text{trans}}}{d\epsilon_{\text{axial}}} = -\frac{d\epsilon_y}{d\epsilon_x} \tag{7}$$

6

where ν is the Poisson's Ratio, ϵ_{trans} is transverse strain and ϵ_{axial} is axial strain. Theoretically, in the case of small deformations Poisson's Ratio ν was computed by Eq. (8) and in the case of large deformation it was computed by Eq. (9).

$$\nu = -\frac{\Delta w/w_0}{\Delta l/l_0} \tag{8}$$

$$\nu = -log_{(1+\Delta l/l_0)}(1 + \Delta w/w) \tag{9}$$

To verify that the spring model can catch isotropic elastic material's Young modulus and Poisson ratio, we carried out a set of simulations by stretching fabric surfaces with different Young's modulus and Poisson ratios. These simulations start with a fabric surface which has its original length $l_0 = 0.1m$ and original width $w_0 = 0.02m$. The fabric is then pulled along the direction of the longer side of it with a distributed force.

Firstly, three groups of simulations which stretch fabric surfaces with Poisson ratios of -0.14, -0.22, and -0.30, respectively are carried out. Each group with five different values of Young's modulus, that is $0.1GPa$, $0.2GPa$, $0.3GPa$, $0.4GPa$, and $0.5GPa$ respectively. The fabric surface is pulled at one end. Young's modulus and Poisson ratio are calculated from the deformation of the surface and the force added on it. The results are summarized by Table 2 which shows that the spring-mass model nicely reproduces the values of Young's modulus and the Poisson ratio from the input.

Next, a group of simulations, stretching the fabric surface whose Young's modulus and Poisson ratio are fixed at $0.5Gpa$ and -0.14 respectively, are carried out. The total number of triangles of these triangulations use in the simulations changes from 1143, 1590, 2468, to 4520. The strain of the elongation change from 0.002, 0.004, 0.006, 0.008 to 0.01. The measured Young's modulus and Poisson ratio from these simulations are demonstrated in Figure 5.

The numerical solutions imply that the revised spring-mass model based on the derivation by Delingette [2] can accurately simulate an isotropic elastic membrane in the linear region with strain up to 0.01. The error between the spring model and continuum model increases when elongation is too large and the deformation reaches the nonlinear regime. For parachute simulation, the strain of the canopy, even during the most dynamic phase of inflation, should still be in the linear regime. Therefore the spring model is an excellent model for such simulations.

It should be mentioned that in the case in which the fabric surface is compressed and it adjusts itself with wrinkles, the convergence is not obvious. In some cases, a small perturbation of the initial condition may result in substantially different folding and wrinkling patterns. How to describe such case and define its mathematical convergence remain as an open question to the model.

1.1.4 Fluid Canopy Coupling

To model the parachute system, an accurate coupling between the Naiver-Stokes equation and the structure dynamics must be carefully considered near the canopy surface. The

Group 1	Young's Modulus results (GPa)		Poisson Ratio results	
	Input	Numerical	Input	Numerical
case 1	0.1	0.0963016606	-0.14	-0.141887
case 2	0.2	0.1926061554	-0.14	-0.141920
case 3	0.3	0.2888656609	-0.14	-0.141910
case 4	0.4	0.3852341348	-0.14	-0.141913
case 5	0.5	0.4815205552	-0.14	-0.141905
Group 2	Young's Modulus results (GPa)		Poisson Ratio results	
	Input	Numerical	Input	Numerical
case 1	0.1	0.0957361901	-0.22	-0.223342
case 2	0.2	0.1914918931	-0.22	-0.223332
case 3	0.3	0.2870650506	-0.22	-0.223382
case 4	0.4	0.3829895385	-0.22	-0.223349
case 5	0.5	0.4784196259	-0.22	-0.223365
Group 3	Young's Modulus results (GPa)		Poisson Ratio results	
	Input	Numerical	Input	Numerical
case 1	0.1	0.0953452395	-0.30	-0.305362
case 2	0.2	0.1906619377	-0.30	-0.305342
case 3	0.3	0.2860538596	-0.30	-0.305343
case 4	0.4	0.3814011184	-0.30	-0.305359
case 5	0.5	0.4767297379	-0.30	-0.305355

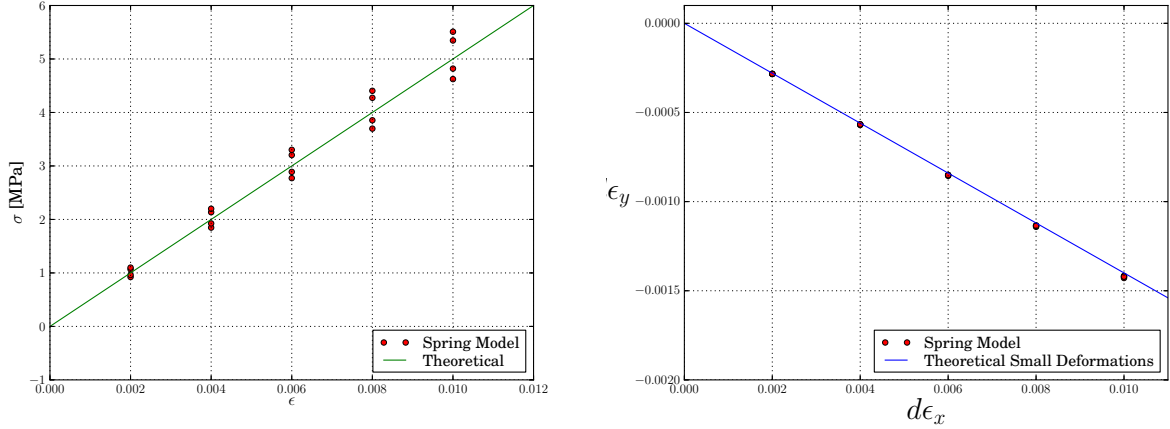Table 2: Young's modulus and Poisson ratio verification

Figure 5: Young's Modulus (Left) and Poisson's Ratio (Right). We tested the spring model by stretching the fabric surface to different lengths. The numerical results show that the spring-mass model catches the fabric's Young's Modulus and Poisson ratio nicely in the linear regime of strain.

method we designed for the simulations of air delivery system uses the superposition of impulse on every mass point. Each mass point in the spring system acts as an elastic boundary point and exerts an impulse on the fluid in its normal direction. Our algorithm ensures that the action and reaction between the spring-mass point and the fluid solver are equal in magnitude and opposite in directions, a requirement of Newton's third law.

We applied Peskin's delta function immersed boundary method [10, 12]

$$\mathbf{f}(\mathbf{x}, t) = \int \mathbf{F}(s, t)\delta(\mathbf{x} - \mathbf{X}(s, t))ds. \tag{10}$$

The difference between our method and Kim and Peskin's method lies in the calculation of $\mathbf{F}$. Instead of computing the tension through the derivative with respect to the arc length, we use the impulse of the mass point as a result of the superposition of three forces from the spring system, that is

$$\mathbf{F}(\mathbf{x}_i, i) = d(\mathbf{I}_g + \mathbf{I}_p + \mathbf{I}_s)/dt \tag{11}$$

Eq. (11) is more physically realistic, especially because $\mathbf{I}_s$ is obtained from the spring equations. In the canopy spring system, we have observed that the tension is high at the top of the canopy where the curvature is almost zero.

1.2 The $k - \varepsilon$ Turbulence Model

A round parachute canopy poses several challenging problems for the understanding of bluff-body wake, which has been studied extensively during the past decades [18, 19, 33, 8]. However, the main difference between fabric canopies and rigid bluff bodies are fabric flexibility and permeability [5]. Thus the coupling of the vertical flow with the permeable and flexible fabric may have significant difference with rigid bluff bodies. Although direct numerical simulation (DNS) and large eddy simulation (LES) may give a relatively accurate prediction of turbulence behavior, the high computational cost are still obstacles of extending these methods to high Reynolds number, which typically exceeds several millions during parachute inflation. Therefore, eddy viscosity models based on the Reynolds Averaged Navier-Stokes (RANS) equations are commonly used in turbulence simulation. One of the most popular RANS-based turbulence model is the $k - \varepsilon$ family, which automatically provide the turbulence length scale or its equivalent and are thus complete [43]. This kind of model still involved some empirical constants, which are widely used and fitted well with some experimental data.

In this section, we firstly give a brief introduction to the $k - \varepsilon$ family, including standard, RNG and realizable model. Then a series of simulations were performed to make a comparison between these models.

1.2.1 Mathematical Formulation

Following the Boussinesq assumption [43], the hydrodynamic behavior of a turbulent incompressible fluid is governed by the RANS equations for the mean velocity U and pressure

p

$$\frac{\partial U}{\partial t} + U \cdot \nabla U = -\nabla p + \nabla \cdot ((\nu + \nu_T)[\nabla U + \nabla U^T]) \tag{12}$$
$$\nabla \cdot U = 0$$

where ν is the kinematic viscosity and ν_T is the turbulent eddy viscosity which is supposed to approximate the turbulent fluctuations. In standard $k - \varepsilon$ model, the eddy viscosity is defined as

$$\nu_T = C_\mu \frac{k^2}{\varepsilon}, \tag{13}$$

where k is the turbulence kinetic energy and ε is the dissipation rate. To compute k and ε, two additional convection-diffusion-reaction equations are defined as:

$$\frac{\partial k}{\partial t} + \nabla \cdot (kU - (\nu + \frac{\nu_T}{\delta_k})\nabla k) = P_k - \varepsilon \tag{14}$$

$$\frac{\partial \varepsilon}{\partial t} + \nabla \cdot (\varepsilon U - (\nu + \frac{\nu_T}{\delta_\varepsilon})\nabla \varepsilon) = \frac{\varepsilon}{k}(C_1 P_k - C_2 \varepsilon) \tag{15}$$

where $P_k = \frac{\nu_T}{2}|\nabla U + \nabla U^T|^2$ is the production of turbulent kinetic energy. For the standard $k - \varepsilon$ model, the default values of the involved empirical constants are: $C_\mu = 0.09$, $C_1 = 1.44$, $C_2 = 1.92$, $\delta_k = 1.0$, $\delta_\varepsilon = 1.3$. Although simple and efficient, the standard model is unable to capture the effects of smaller scales of motion due to its single turbulence length scale. In order to account for the different scales of motion, a mathematical technique, Re-Normalization Group (RNG) method, was used to derive a turbulence model similar to the standard one, results in a modified form of the ε equation:

$$\frac{\partial \varepsilon}{\partial t} + \nabla \cdot (\varepsilon U - (\nu + \frac{\nu_T}{\delta_\varepsilon})\nabla \varepsilon) = \frac{\varepsilon}{k}(C_1 P_k - C_2^* \varepsilon) \tag{16}$$

$$C_2^* = C_2 + \frac{C_\mu \eta^3 (1 - \eta/\eta_0)}{1 + \beta \eta^3} \tag{17}$$

$\eta = kS/\varepsilon$, S is modulus of the mean rate of strain tensor. The coefficients are derived explicitly in the RNG procedure and are also listed here for completeness: $C_\mu = 0.0845$, $C_1 = 1.42$, $C_2 = 1.68$, $\delta_k = 0.7194$, $\delta_\varepsilon = 0.7194$ [44].

Using standard or RNG model may produce unphysical results when the mean strain rate is large. The most straightforward way to ensure the realizability is to make C_μ variable according to the mean flow. It is around 0.05 in homogeneous shear flow and around 0.09 in the inertial sub-layer of boundary layer flow. The kinetic energy equation remains the same as standard model, while the equation for dissipation rate is modified as follows:

$$\frac{\partial \varepsilon}{\partial t} + \nabla \cdot (\varepsilon U - (\nu + \frac{\nu_T}{\delta_\varepsilon})\nabla \varepsilon) = \frac{\varepsilon}{k}(C_1 S - C_2 \frac{\varepsilon}{k + \sqrt{(\nu \varepsilon)}}) \tag{18}$$

$$C_1 = \max\{0.43, \frac{\eta}{\eta + 5}\}, \eta = S\frac{k}{\varepsilon} \tag{19}$$

The coefficient C_μ is determined to meet the positive normal stress and Schwarz's inequalities. The details of implementing Realizable model can be found in [23] and the reference therein.

The implementation of $k - \varepsilon$ model has no difference with solving other convection-diffusion-reaction equations, except that two things need to be noticed. One must specify an appropriate boundary condition to a solid wall for the velocity and two turbulence parameters. Due to the extremely high Reynolds number in parachute simulation, it is worthwhile to use the wall functions to bridge the viscosity-affected region and the fully-turbulent region avoiding the need for resolution of strong velocity gradients [15]. Another task is to avoid loss of positivity of k and ε due to computational errors. This can be achieved by simply keeping the coefficients positive for the linearized equations without touching any primitive variables [16]. In the next subsection, a series of parachute simulation are performed to compare the results between these three different models.

1.2.2 Numerical Implementation and Test Results

In this section, we examine incompressible flow past a canopy surface placed in a channel orthogonal to the streamwise fluid. The numerical domain is set to be $12m \times 24m$ with Dirichlet boundary conditions in the streamwise while slip, no penetration boundary conditions are applied at the channel walls and canopy surfaces. The configurations are illustrated in Figure 6. The Reynolds number is defined as:

$$Re = \frac{UL}{\nu} \tag{20}$$

where $U = 3ms^{-1}$ is the inflow velocity, $L = 7.0m$ is the width of the canopy, $\nu = 1.5 \times 10^{-5}m^2s^{-1}$ is the kinematic viscosity of the air, thus $Re \approx 10^6$.

We use finite difference method to solve the RANS equations. The pressure and velocity in equation (12) are decoupled with Chorin's projection method and advanced in small time step according to 4-th order Runge-Kutta method with numerical flux evaluated by 5-th order WENO scheme. The equations for k and ε are discretized with Crank-Nicolson scheme to achieve second order of accuracy. Some numerical techniques similar to [16] are applied to preserve the positivity of turbulence parameters.

We show the numerical results by applying different turbulence models. Figure 7 and Figure 8 display the viscosity and velocity computed from different models. The RNG and realizable model perform better than the standard one since they successfully capture the separation and rotation of the air flow.

1.3   Numerical Implementation of Porosity

Fabric porosity of parachute canopy greatly affects parachute performance. It has long been known that fabric permeability is an important factor in parachute design. By carefully considering the balance between payload and drag, a finite permeability will make the parachute much more stable. The permeability of the parachute material often plays a vital role in
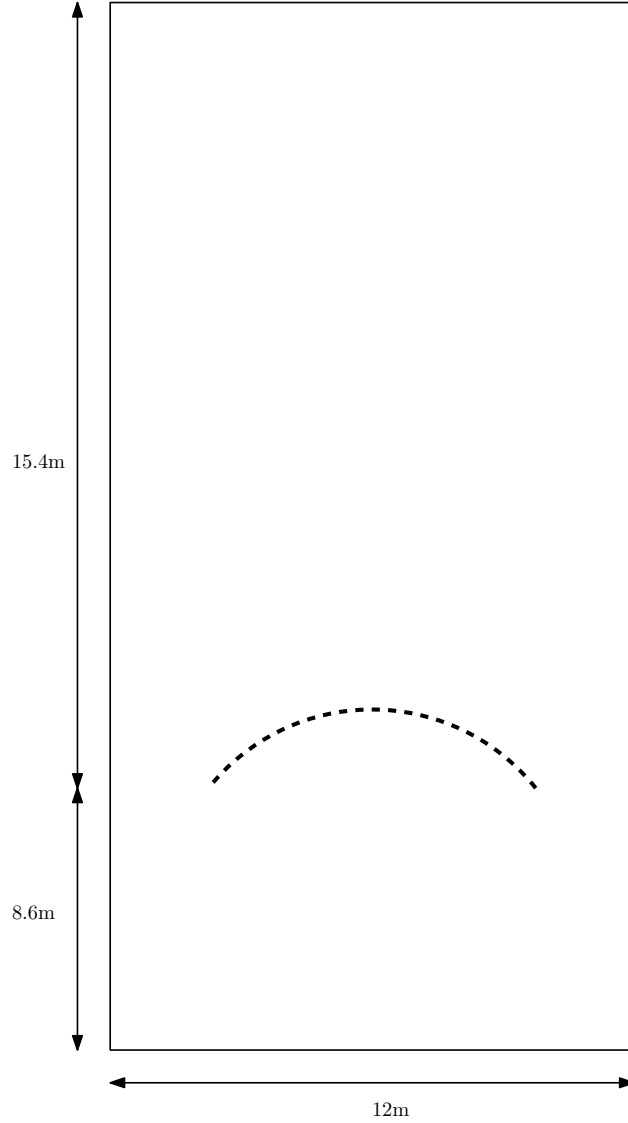
Figure 6: Experimental settings

the design of parachute. A state-of-art tuning between an impervious material to a highly permeable fabric can make a parachute from a wandering sloth into a plummeting stabilizer.

There are two forms of parachute porosity: for canopy manufactured from solid fabric, the porosity (sometimes referred as fabric permeability) is defined as the airflow through canopy cloth in $ft^3/ft^2/min$ (cubic feet per minute per square feet) at $\frac{1}{2}$ inch water pressure; for slotted canopies which has geometric openings, porosity is defined in percent as the ratio of all open areas to the total canopy area. Most personnel parachutes and the main descent parachutes for air vehicles use materials with porosity values from 80 to 150 $ft^3/ft^2/min$. Gliding parachutes use almost imporous materials from 0 to 5 $ft^3/ft^2/min$. Slotted parachutes use geometric porosity values in 10% to 35% range [14].
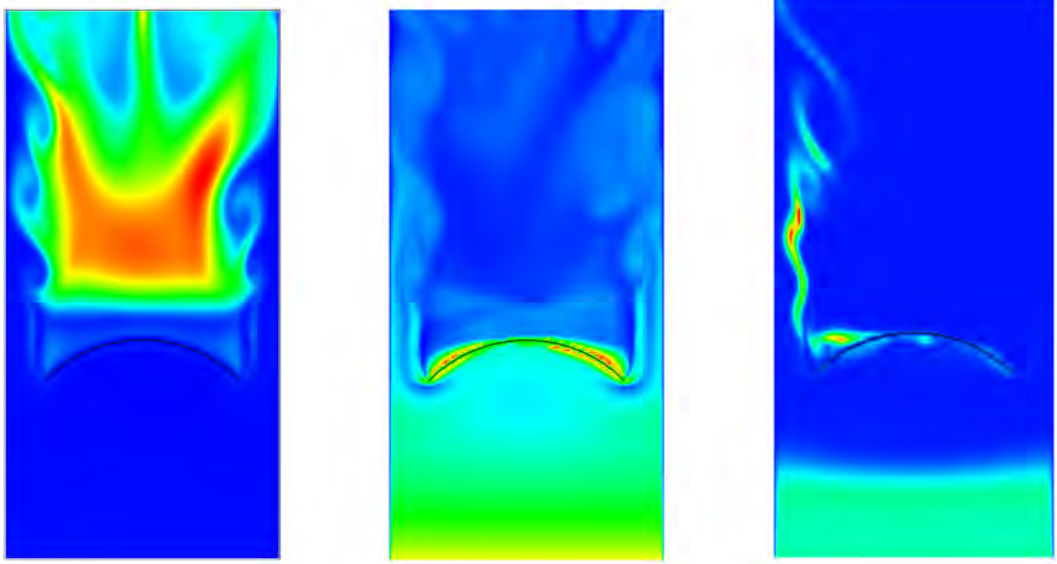
Figure 7: Eddy viscosity from different models at $t = 10s$, from left to right are standard, RNG and realizable $k - \varepsilon$ model

Porosity influences parachute drag, stability and opening forces. Higher porosity decreases opening forces and oscillation, but reduces drag forces at the same time which is usually not desirable. As a result, parachute canopy porosity is an important factor to consider for both parachute design and parachute simulation. Different types of parachutes have different design characteristics requirements [4, 14, 22].

Given the significance of porosity, there are several attempts to simulate porous canopy parachute motion. Kim and Peskin [11] use immersed boundary method to simulate parachute motion and derive the relative velocity between fluid and canopy interface using Darcy's Law. Tutt [39] simulated parachute performance under LS-DYNA and use Ergun equation to describes the magnitude of porous flow. Wang, Aquelet, Tutt, Do, Chen and Souli [42] simulate the interaction between the fluid and porous medium by a Euler-Lagrange coupling under LS-DYNA framework.

For Kim and Peskin's method, porosity is considered as the leaking pores in the immersed elastic boundary. Let β be the density of pores, which means there are βds pores in the interval $(s, s + ds)$ along the surface. If each pore has an aerodynamic conductance equal to γ, which means the flux through the pore is $\gamma(p_1 - p_2)$ where p_1 and p_2 are the pressures on the two sides of the boundary, then the flux through the interval $(s, s + ds)$ of the boundary is given by $\beta\gamma(p_1 - p_2)ds$.

Tutt's algorithm is closest to the front tracking implementation. Tutt used the Ergun equation to describe the magnitude of porous flow velocity at a given pressure difference
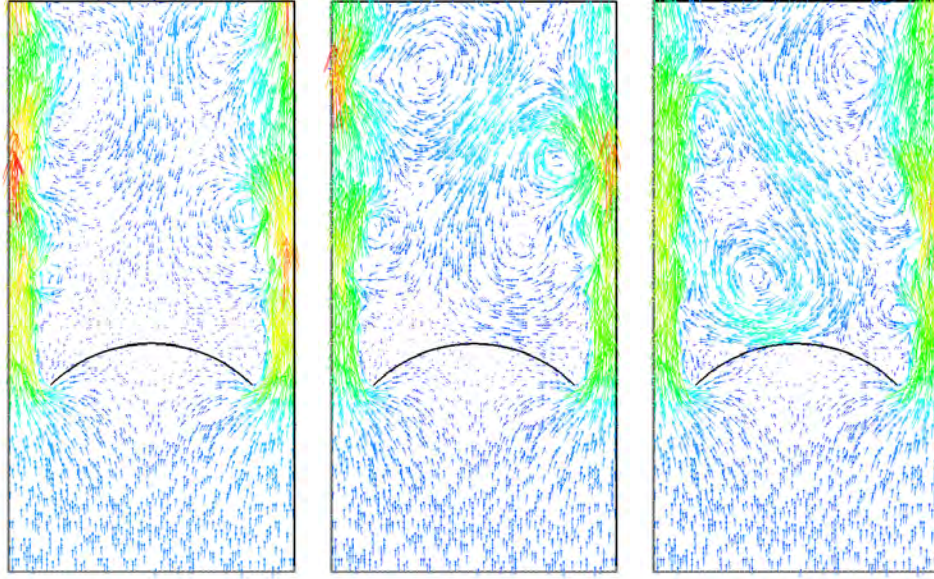
Figure 8: Velocity field from different models at $t = 10s$, from left to right are standard, RNG and realizable $k - \varepsilon$ model with color indicating the magnitude of velocity

based on two coefficients in the following equation:

$$\frac{\Delta P}{L} = \frac{\mu}{K_1} \cdot v_f + \frac{\rho}{K_2} \cdot v_f^2 = a \cdot v_f + b \cdot v_f^2$$

where:

$$K_1 = \frac{\varepsilon^3 \cdot D^2}{150 \cdot (1 - \varepsilon)^2}$$

$$K_2 = \frac{\varepsilon^3 \cdot D}{1.75 \cdot (1 - \varepsilon)}$$

are referred to as the viscous and inertial factors respectively. D is defined as the characteristic length, ε is the porosity and is equal to the ratio of the void and total volume. v_f, μ and ρ are fluid velocity, viscosity and density respectively. In our FronTier-airfoil code, the pressure difference is computed after projecting the velocity into the divergence-free space. We can then use the pressure difference on two sides of the canopy to compute the v_f and add it to the advection solver as flux from the immersed elastic surface.

A cloth porosity of 27.4 $ft^3/ft^2/min$ at $\frac{1}{2}$ inch water pressure is equivalent to 1% geometric porosity [14]. This enables comparison between different parachutes and uniform simulations among parachutes which have different porosity types.

We introduce the concept of penetration ratio γ, which is a dimensionless coefficient $0 \leq \gamma \leq 1$. When $\gamma = 0$, it means no penetration will happen and the fluid on two sides of boundary has no connection; when $\gamma = 1$, it means the interface does not exit; when

$0 < \gamma < 1$, part of the fluid can penetrate the canopy and canopy interface is a porous medium. The fluid field is described by Navier-Stokes equation and solved by projection method [1]. Coupling penetration ratio into projection method transforms fluid through porous medium simulation into boundary condition treatment.

The projection method contains two major steps: transportation process to calculate intermediate velocity and projection process to derive pressure and new velocity.
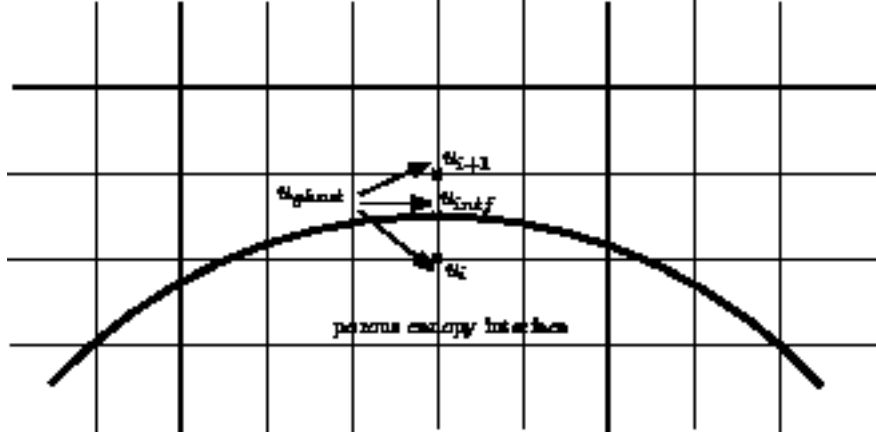

Figure 9: Ghost Point on Canopy Interface

In the advection part, fifth-order WENO scheme is used to proceed calculation which requires information from seven nearby points, i.e.

$$u_i^{k+1} = f\left(u_{i-3}^k, u_{i-2}^k, u_{i-1}^k, u_i^k, u_{i+1}^k, u_{i+2}^k, u_{i+3}^k\right).$$

When came across boundary, interpolation is required to obtain ghost points. Assume parachute canopy is between u_i and u_{i+1}, then $u_{i+1}^k, u_{i+2}^k, u_{i+3}^k$ should be interpolated as

$$u_{i+1}^{k,ghost} = g_1\left(\left\{u_{1,\cdots,n}^k\right\}\right),$$

$$u_{i+2}^{k,ghost} = g_2\left(\left\{u_{1,\cdots,n}^k\right\}\right),$$

$$u_{i+3}^{k,ghost} = g_3\left(\left\{u_{1,\cdots,n}^k\right\}\right).$$

Thus, the calculation of velocity on the boundary is

$$u_i^{k+1} = f\left(u_{i-3}^k, u_{i-2}^k, u_{i-1}^k, u_i^k, u_{i+1}^{k,ghost}, u_{i+2}^{k,ghost}, u_{i+3}^{k,ghost}\right).$$

Coupling penetration ratio γ, we set boundary points as

$$u_{i+1}^{k,poro} = (1-\gamma)\, u_{i+1}^k + \gamma u_{i+1}^{k,ghost}$$

$$u_{i+2}^{k,poro} = (1-\gamma)\, u_{i+2}^k + \gamma u_{i+2}^{k,ghost}$$

15

$$u_{i+3}^{k,poro} = (1 - \gamma) u_{i+3}^{k} + \gamma u_{i+3}^{k,ghost}$$

which leads to new construction formula

$$u_i^{k+1} = f\left(u_{i-3}^k, u_{i-2}^k, u_{i-1}^k, u_i^k, u_{i+1}^{k,poro}, u_{i+2}^{k,poro}, u_{i+3}^{k,poro}\right).$$

Diffusion part is solved using Crank-Nicolson scheme which has common form

$$\frac{u_i^{k+1} - u_i^k}{\triangle t} = \frac{1}{2}\left(\frac{u_{i+1}^{k+1} - 2u_i^{k+1} + u_{i-1}^{k+1}}{\triangle x^2} + \frac{u_{i+1}^k - 2u_i^k + u_{i-1}^k}{\triangle x^2}\right)$$

and given $a = \frac{\triangle t}{2\triangle x^2}$, it can be simplified as

$$-au_{i+1}^{k+1} + (1 + 2a) u_i^{k+1} - au_{i-1}^{k+1} = au_{i+1}^k + (1 - 2a) u_i^k + au_{i-1}^k.$$

Assume there is canopy interface between u_i and u_{i+1}, and consider no porosity case, we get the velocity from interface $u_{i+1}^{k,state}$. Coupling with constant velocity boundary condition, we have

$$-au_{i+1}^{k,state} + (1 + 2a) u_i^{k+1} - au_{i-1}^{k+1} = au_{i+1}^{k,state} + (1 - 2a) u_i^k + au_{i-1}^k.$$

Introducing penetration ratio gives

$$-\gamma au_{i+1}^{k+1} + (1 + 2a) u_i^{k+1} - au_{i-1}^{k+1} = 2(1 - \gamma) au_{i+1}^{k,state} + \gamma au_{i+1}^k + (1 - 2a) u_i^k + au_{i-1}^k.$$

After obtaining intermediate velocity, projection step is carried out to obtain pressure and new velocities. Solving Poisson equation gives pressure

$$p_{i+1} - 2p_i + p_{i-1} = \frac{\rho \triangle x^2}{\triangle t} div\left(u_i^*\right).$$

Considering the case which has canopy interface between u_i and u_{i+1}, and using Neumann boundary will give the formula

$$-p_i + p_{i-1} = \frac{\rho \triangle x^2}{\triangle t} div\left(u_i^*\right).$$

Introducing penetration ratio γ gives

$$\gamma p_{i+1} - (1 + \gamma) p_i + \gamma p_{i-1} = \frac{\rho \triangle x^2}{\triangle t} div\left(u_i^*\right).$$

Update new velocity uses

$$u_i^{k+1} = u_i^* - \frac{\triangle t}{2\rho \triangle x}\left(p_{i+1} - p_{i-1}\right)$$

16

when came across boundary between u_i and u_{i+1}, the new formula coupled with penetration ratio is

$$u_i^{k+1} = u_i^* - \frac{\gamma \triangle t}{2\rho \triangle x} \left(p_{i+1} - p_{i-1} \right).$$

The penetration ratio γ is a dimensionless parameter which controls the volume of fluid going through canopy via influencing boundary condition. In the case of zero porosity, the boundary conditions used for intermediate velocity and pressure in transportation and projection process respectively are

$$u_b^* = u_{intf}$$

$$\nabla p_b = 0$$

where u_{intf} is the velocity of interface, i.e. parachute canopy. This gives the boundary condition of fluid at interface through projection formula

$$u^* = u + \frac{\triangle t}{\rho} \nabla p \Rightarrow u_b = u_{intf}$$

which means the velocity of fluid at interface is the velocity of the interface.

After coupling penetration ratio, the boundary condition of intermediate velocity is derived from interpolation

$$u_b^* = \begin{cases} \left(1 - \gamma\right) u_{intf} + \gamma u_{i+1}, & \text{for } u_i \\ \left(1 - \gamma\right) u_{intf} + \gamma u_i, & \text{for } u_{i+1} \end{cases}$$

and the boundary condition for pressure is

$$\nabla \tilde{p}_b = \frac{\tilde{p}_{i+1} - p_i}{\triangle x} = \gamma \nabla p_b.$$

Substitute above equations into projection formula gives

$$u_b = \begin{cases} u_{intf} + \gamma \left(u_{i+1} - u_{intf} \right) + \gamma \frac{\triangle t}{\rho} \nabla p_b, & \text{for } u_i \\ u_{intf} + \gamma \left(u_i - u_{intf} \right) + \gamma \frac{\triangle t}{\rho} \nabla p_b, & \text{for } u_{i+1} \end{cases}.$$

The difference between the fluid velocity at boundary and the boundary interface velocity is regarded as velocity due to porosity, which can be represented as

$$u_{poro} = -\gamma \frac{\triangle x^2}{2} \Delta u_b - \gamma \frac{\triangle t}{\rho} \nabla p_b.$$

Darcy's law describes the fluid through porous medium, which has the form

$$q = -\frac{\kappa}{\mu} \nabla p$$

17

where ∇p is the pressure gradient, κ is the intrinsic permeability of medium, μ is the fluid viscosity and q is the flux. The fluid velocity is related to flux q by void fraction ϕ

$$u_{poro} = \frac{q}{\phi}.$$

Equating the fluid velocity derived from numerical scheme with Darcy's law gives

$$\gamma = \frac{\frac{\kappa}{\lambda\mu}\nabla p}{\frac{\triangle t}{\rho}\nabla p + \frac{\triangle x^2}{2}\triangle u}$$

assume $\triangle x^2 \ll \triangle t$ gives simplified form

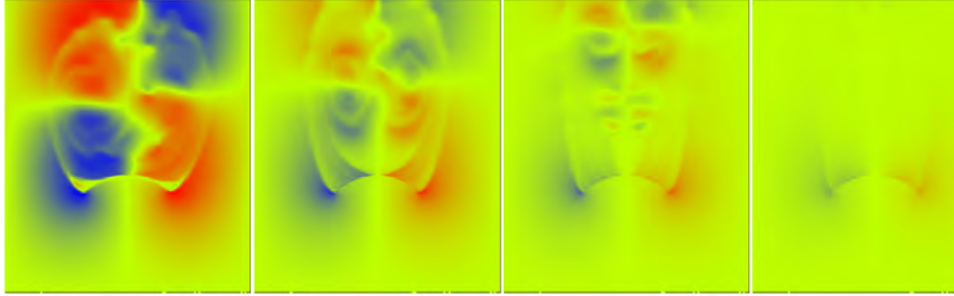$$\gamma \approx \frac{\kappa\rho}{\triangle t\phi\mu}.$$

Figure 10: Velocity in x direction, from left to right: $\gamma = 0, 0.25, 0.5, 0.75$
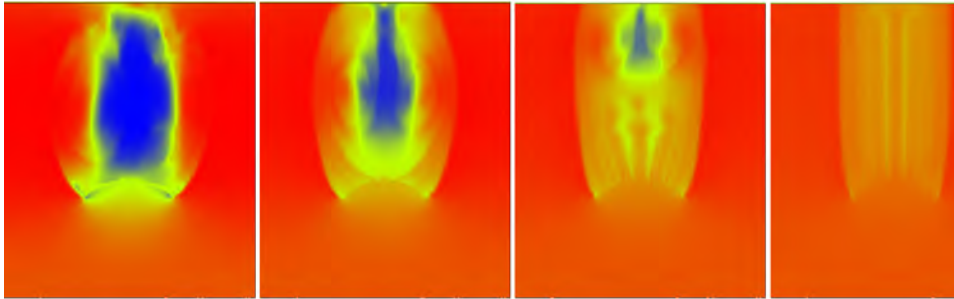

Figure 11: Velocity in y direction, from left to right: $\gamma = 0, 0.25, 0.5, 0.75$

1.4 Parachute Simulation and Code Validation

Our primary objective is to conduct the fabric-fluid coupled simulations and validate it against the available parachute experiments. Parachutes differ in geometry and dimensions

Figure 12: Vorticity, from left to right: $\gamma = 0, 0.25, 0.5, 0.75$

of the canopies and risers. Among these parachutes, the T-10 parachute is used by the Army as a personnel carrier. This type of parachute has a parabolic shape for its canopy with a vent at the top. The G-11 parachute is a cargo parachute with no vent. Its dimensions could vary and is usually used as a multiple parachute system to deliver supporting equipment. We have studied the single 1/3-scale G-11 parachute inflation. The cross parachute has four open side vents and can be used as a sports parachute and small cargo carrier. Figure 13 shows the inflation sequence of a cross parachute. These three parachutes produce different patterns of airflow around the canopy and exert different drags to the parachute system.

For comparison with the indoor vertical parachute test, we initialized the parachute with flat circular canopy of 2.134 m (7 ft) nominal diameter. We carried out pre-step running to deform the parachute canopy and then used the resulting geometry as the initial state for the drop test simulation. The initial skirt diameter makes different time graph for full inflation. In this test, there is a breathing motion due to the over-inflation of the canopy. It is shown that the canopy progresses into an over-inflated shape, almost flat, and then contracts and over-inflates again. The breathing motion is an oscillatory motion. The canopy appeared to expel the excess air by means of the breathing. In the experiment, the breathing motion was also caused by the constraint on the parachute, imposed by the guide wire. The breathing motion in the simulation is smaller because there is no vertical motion restriction such as a guide wire. The terminal velocity of canopy has a good agreement with the vertical parachute test results. The experimental data shows that the descent speed rises rapidly to a peak. It slows down while the parachute inflates and then slowly approaches a steady descent speed of 4.27 m/s (14 ft/s). Even though the numerical solution cannot be compared with the experimental data during the initial period of time in the simulation, we have observed that it reaches the terminal velocity of about 3.9 m/s. The difference between the terminal velocity in numerical simulation and the experimental data is thought due to the lack of porosity of the canopy in the current numerical model. We also simulated the 1/3-scale G-11 parachute with different canopy folding level, they all converge to the terminal speed of 3 m/s in several seconds.

Parachute breathing is an oscillatory motion caused by the interaction between the fluid force and the canopy at the skirt of the parachute. The large difference between upper and lower side pressure causes vibration in the projected drag area of the canopy and thus

19

the oscillation in descent speed. Numerical study of breathing frequency is important for understanding the stability of the parachute. In our simulation, we initialized 2.134 m (7 ft) flat-circular parachute to compare with experimental data by Tutt *et al.*[40]. The breathing frequency in our simulation is approximately 1.6 - 2.0 Hz (0.5 s to 0.6 s period). This is in the acceptable range with the experimental frequency which is 2.0 Hz (0.5 s period). If we use a parachute with larger size or lower Reynolds number, the breathing period will increase. For example, the average period of breathing for the 10.7 m T-10 parachute is 2.3 s in experiment [6], while it is about 2 s in our simulation.
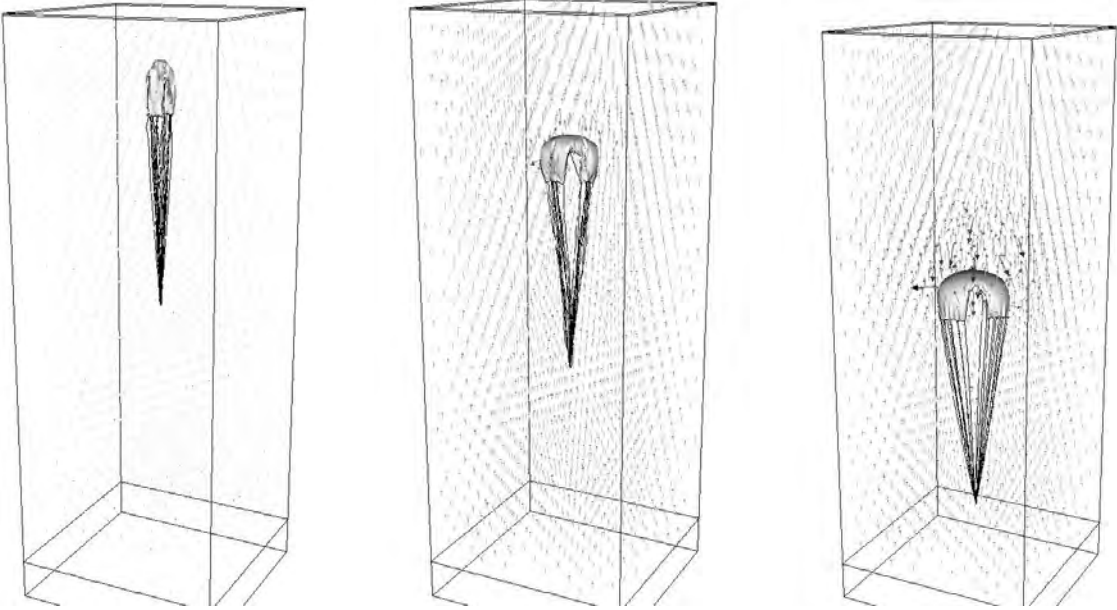

Figure 13: Simulation of cross parachute unfolding and inflation. The starting state of the parachute is partially folded state. The parachute has a small horizontal drift because the folded state is not perfect symmetric.

1.5   Other Computational Development

In the last year, we have accomplished several new tasks in the parachute simulation code which enabled us for more realistic and robust computation on the parachute problem.

GPU Accelerated Spring Module Graphics Processing Unit (GPU) computing [13] is to use the GPUs together with CPUs to accelerate a general-purpose scientific and engineering application. GPU computing can offer dramatically enhanced application performance by offloading computation-intensive portions of the programming code to the GPU units, while the remainder of the code still runs on the CPU. Joint CPU/GPU applications constitute a powerful combination because CPUs consist of a few cores optimized for serial processing, while GPUs consist of thousands of smaller, more efficient cores designed for
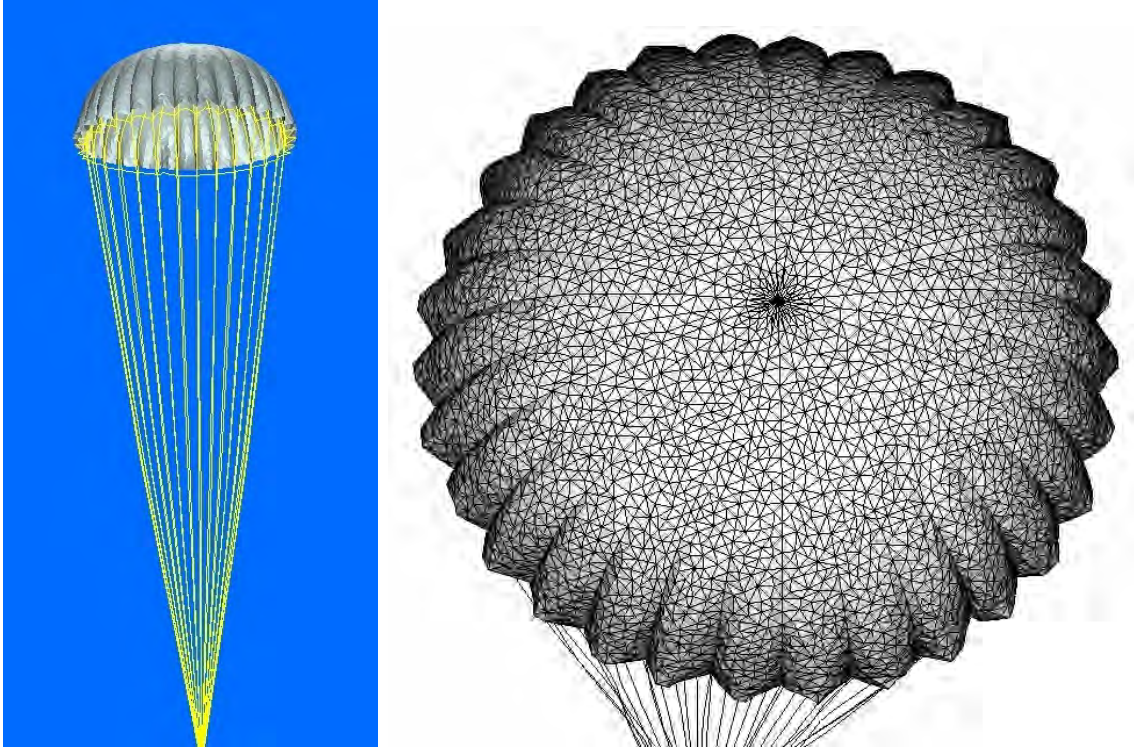
Figure 14: The reinforcement cables, or the gores, are modeled using the same spring system, but with different spring constant along the gore curves. The left plot shows fully opened canopy with gores. After the inflation, the gore structure is clearly revealed. The right plot shows the surface mesh and the details of the gore curves.
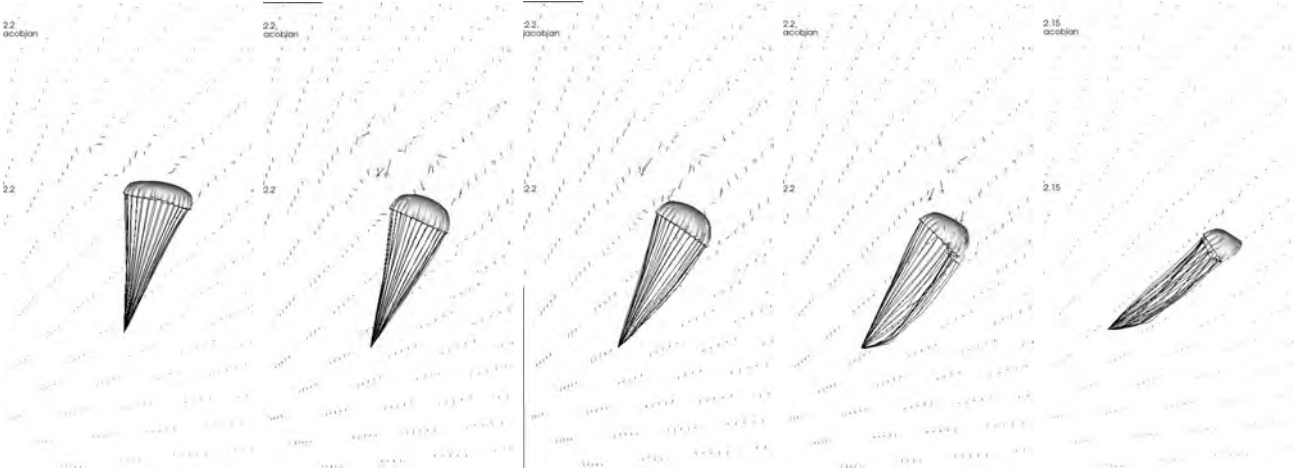

Figure 15: Angled deployment of C-9 parachute with the flow. The deployment angle between the initial parachute and the direction of flow are (from left to right) 15°, 30°, 45°, 60°, 75°, respectively.

massive parallel calculations. Serial portions of the code with logical comparison run on the CPU while floating point operation intensive parallel portions of the code run on the GPU. The total operation time recorded by the CPU clock and time for GPU intensive computational part are collected in Table 3 in seconds. From Table 3 we can conclude that the application of the GPU has a clear advantage in the computation of spring model. To fully take advantage of the GPU system, further optimization are needed to the fluid solver. Table 4 shows the hardware structure of our computer.

Mesh	Total Time (sec)		Per Step (sec)	
	Without-GPU	With-GPU	CPU	GPU
$80 \times 80 \times 20$	1345	152	1.71	0.12
$160 \times 160 \times 40$	8115	804	5.02	0.25
$320 \times 320 \times 80$	40291	5349	11.63	0.55

Table 3: GPU and CPU computing time of three dimensional spring model. Spring models with three different mesh sizes ($80 \times 80 \times 20$, $160 \times 160 \times 40$, $320 \times 320 \times 80$) were tested with both pure CPU code and hybrid (CPU and GPU) code. The hybrid code is 15-20 times faster than the pure CPU code for the computation intensive part. In general, the overall speed of the hybrid code is 7-10 times faster than the pure CPU code for the spring-mass model part.

Hardware	CPU	Dual Eight Core XEON E5-2687W, 3.1GHz
		64GB DDR3
		32KB x 16 L1 Cache, 256KB x 16 L2 Cache, 20MB x 2 L3 Cache
	GPU	Dual Quadro 6000 with 14 multiprocessor, 448 cores, 1.15Hz
		6GB global memory, 64 KB constant memory
		48KB shared memory and 32768 registers per multi-processor
Software	OS	Fedora 18 with kernel 3.9.2-200.fc18.x86_64
	Compiler	gcc version 4.7.2
	CUDA	CUDA Toolkit 5.0

Table 4: A Dell Precision T7600 Workstation with dual NVIDIA Quadro graphics cards was used to set up the test environment.

Parachute gores are stitched by the reinforcement cables. The reinforcement cables are important structures on the parachute canopy surface. To accurately model the aerodynamic motion of a parachute, we need to create a mathematical model which reveals the geometry as well as the material strength of the canopy surface and the gores. The gore structures in the parachute system have important effect on the stability of the parachute motion. In our model, we treat the reinforcement cables as curves embedded in the canopy surface.

Young's modulus for the fabric surface and the reinforcement cable are set at different values. This is realized by assigning different spring constants to the surface mesh and to the gore curves. The insertion of gores as stiffened interior curves in the canopy surface mesh is demonstrated by Figure 1. Figure 14 shows that when fully inflated, the spring system reveals both the patches and the indentation of the reinforcement cables. The effect of the gores on suppressing the vortex flow at the top of the canopy is being studied. This will be published in a new paper we are currently working on.

Collision handling on parachute contact is through the modified *FronTier* library. The computational module for parachute simulation is built on the *FronTier* library. We have used many existing data structures and functionality for geometrical handling. However, the study of parachute, airfoil and other fluid structure based simulations requires major revisions to the software library. Among those needed are functions to handle the non-manifold surface and three dimensional curves (not as boundary of a surface, such as the string chords). The original *FronTier* code was designed for the study of fluid interface instability problems. For these problems, collision surfaces are resolved by merging or bifurcation. However, fabric surface can neither merge nor bifurcate, therefore we need to have functions which can carefully deal with the repulsive contact and collision.

Global indexing is a new feature added to the parachute code. The original *FronTier* had to deal with frequent surface mesh optimization and topological reconstructions. This makes the parallelization based on global index very difficult. As a result, the original *FronTier* library relied on floating point matching for parallel communication. The floating point matching is not 100 percent reliable and some more complicated algorithms have to be implemented as reinforcement. However for fabric surface, especially when spring model is used, the inter-connectivity and proximity of the interface marker points are not changed. Therefore, global indexing is ideal for the parallel communication of the surface. We have added a new parallel code for this functionality.

Modularization is emphasized in our code development. The parachute module is an independent application program. This new module consists of four components: (1) the initialization module, (2) the ODE module for the spring system, (3) the PDE module for fluid dynamics, and (4) the *FronTier* library for the interface geometry handling. A total of about 15,000 lines of code have been written for the first three component. An additional 4,500 lines have been added to *FronTier* to adapt the library for the computation of fabric surface.

1.6   Impact on Education and Training

The grant W911NF1310291 has partially supported two graduate students, Bernard Moore and Yiyang Yang. One of them (Bernard Moore) is an American student from a lower income family. He is a highly motivated student and is interested in the US defense research after graduation. He participated in the US Air Force SFFP program and is highly motivated by our project as well as the service to the military. Another former student from our research group, Eric Rathnayke has joined Military Accessions Vital to the National Interest

(MAVNI) program this year. The numerical techniques we studied so far have had an impact on education and training.

Yiyang Yang will soon graduate with the Ph. D. degree in computational and applied mathematics. She has already been offered as a software engineer by Goldman Sachs. Another student, Qiangqiang Shi, not supported by this grant but played an important role in this project, has already received his Ph. D. degree and joined Bloomberg as a software engineer.

1.7   Communication and Outreach of the Project

Our project has been constantly advised by Army scientist Dr. Richard Charles with his expertise in modelling, experiment and field experience on the parachute delivery system. One joint paper has been submitted to the Journal of Fluid and Structure which is still under review. We anticipate a second paper with him in the coming months. The PI has also reached Air Force scientist Dr. Daniel Reasor in the Air Force Summer Fellowship program. Other collaborations include a joint research project with Drs. Robert McGraw and Yangang Liu at Brookhaven National Lab on climate modelling and with Dr. Valmor de Almeida at Oak Ridge National Lab on the study of phase transition problems in nuclear reactor.

References

[1] D. L. BROWN, R. CORTEZ, AND M. L. MINION, *Accurate projection methods for the incompressible navier–stokes equations*, Journal of Computational Physics, 168 (2001), pp. 464–499.

[2] H. DELINGETTE, *Triangular springs for modeling nonlinear membranes*, IEEE Transactions on Visualization and Computer Graphics Volume 14 Issue 2, (2008), pp. 723–731.

[3] J. M. GERE, *Mechanics of materials, sixth edition*, Bill Stenquist, 2004.

[4] H. G. HEINRICH AND E. L. HAAK, *Stability and drag of parachutes with varying effective porosity*, tech. rep., DTIC Document, 1971.

[5] H. JOHARI AND K. J. DESABRAIS, *Vortex shedding in the near wake of a parachute canopy*, Journal of Fluid Mechanics, 536 (2005), pp. 185–207.

[6] H. JOHARI AND K. J. DESABRAIS, *Vortex shedding in the near wake of a parachute canopy*, J. Fluid Mech., 536 (2005), pp. 185–207.

[7] K. KARAGIOZIS, R. KAMAKOTI, F. CIRAK, AND C. PANTANO, *A computational study of supersonic disk-gap-band parachutes using large-eddy simulation coupled to a structural membrane*, Journal of Fluids and Structures, 27 (2011), pp. 175–192.

[8] H. Kim and P. Durbin, *Observations of the frequencies in a sphere wake and of drag increase by acoustic excitation*, Physics of Fluids (1958-1988), 31 (1988), pp. 3260–3265.

[9] J.-D. Kim, Y. Li, and X.-L. Li, *Simulation of parachute FSI using the front tracking method*, Journal of Fluids and Structures, 37 (2013), pp. 101–119.

[10] Y. Kim and C. S. Peskin, *2-D parachute simulation by the immersed boundary method*, SIAM J. Sci. Comput., 28 (2006), pp. 2294–2312.

[11] Y. Kim and C. S. Peskin, *2-d parachute simulation by the immersed boundary method*, SIAM Journal on Scientific Computing, 28 (2006), pp. 2294–2312.

[12] Y. Kim and C. S. Peskin, *3-D parachute simulation by the immersed boundary method*, Comput. Fluids, 38 (2009), pp. 1080–1090.

[13] D. B. Kirk and W. H. Wen-mei, *Programming massively parallel processors: a hands-on approach*, Morgan Kaufmann, 2010.

[14] T. W. Knacke, *Parachute recovery systems design manual*, tech. rep., DTIC Document, 1991.

[15] D. Kuzmin, O. Mierka, and S. Turek, *On the implementation of the k-epsilon turbulence model in incompressible flow solvers based on a finite element discretisation*, International Journal of Computing Science and Mathematics, 1 (2007), pp. 193–206.

[16] A. J. Lew, G. C. Buscaglia, and P. M. Carrica, *A note on the numerical treatment of the k-epsilon turbulence model*, International Journal of Computational Fluid Dynamics, 14 (2001), pp. 201–209.

[17] Y. Li, I.-L. Chern, J.-D. Kim, and X.-L. Li, *Numerical method of fabric dynamics using front tracking and spring model*, Communications in Computational Physics, 14 (2013), pp. 1228–1251.

[18] R. Natarajan and A. Acrivos, *The instability of the steady flow past spheres and disks*, Journal of Fluid Mechanics, 254 (1993), pp. 323–344.

[19] H.-P. Pao and T. W. Kao, *Vortex structure in the wake of a sphere*, Physics of Fluids (1958-1988), 20 (1977), pp. 187–191.

[20] J. W. Purvis, *Prediction of line sail during lines-first deployment*, AIAA 21st Aerospace Sciences Meeting, (1983).

[21] ——, *Numerical prediction of deployment, initial fill, and inflation of parachute canopies*, 8th AIAA Aerodynamic Decelerator and Balloon Technology Conference, (1984).

[22] W. Shepardson, *Problems of parachute design and their relation to textiles*, tech. rep., DTIC Document, 1954.

[23] T.-H. Shih, W. W. Liou, A. Shabbir, Z. Yang, and J. Zhu, *A new k-epsilon eddy viscosity model for high reynolds number turbulent flows*, Computers & Fluids, 24 (1995), pp. 227–238.

[24] K. Stein, R. Benney, V. Kalro, T. E. Tezduyar, J. Leonard, and M. Accorsi, *Parachute fluid-structure interactions: 3-D computation*, Comput. Methods Appl. Mech. Engrg, 190 (2000), pp. 373–386.

[25] K. Stein, R. Benney, T. Tezduyar, and J. Potvin, *Fluid-structure interactions of a cross parachute: numerical simulation*, Computer Methods in Applied Mechanics and Engineering, 191 (2001), pp. 673–687.

[26] K. Stein, T. Tezduyar, V. Kumar, S. Sathe, R. Benney, E. Thornburg, C. Kyle, and T. Nonoshita, *Aerodynamic interactions between parachute canopies*, J. Appl. Mech., 70 (2003), pp. 50–57.

[27] K. R. Stein, R. J. Benney, V. Kalro, A. A. Johnson, and T. E. Tezduyar, *Parallel computation of parachute fluid-structure interactions*, 14th Aerodynamic Decelerator Systems Technology Conference, (1997).

[28] K. R. Stein, R. J. Benney, E. C. Steeves, D. U.S. Army Natick Research, and E. Center, *A computational model that couples aerodynamic and structural dynamic behavior of parachutes during the opening process*, Technical report (U.S. Army Natick Laboratories), United States Army Natick Research, Development and Engineering Center, Aero-Mechanical Engineering Directorate, 1993.

[29] K. R. Stein, R. J. Benney, T. E. Tezduyar, J. W. Leonard, and M. L. Accorsi, *Fluid-structure interactions of a round parachute: Modeling and simulation techniques*, J. Aircraft, 38 (2001), pp. 800–808.

[30] J. H. Strickland, V. L. Porter, G. F. Homicz, and A. A. Gossler, *Fluid-structure coupling for lightweight flexible bodies*, 17th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar, (2003).

[31] K. Takizawa, C. Moorman, S. Wright, T. Spielman, and T. E. Tezduyar, *Fluid-structure interaction modeling and performance analysis of the orion spacecraft parachutes*, International Journal for Numerical Methods in Fluids, 65 (2011), pp. 271–285.

[32] K. Takizawa, T. Spielman, and T. E. Tezduyar, *Space-time FSI modeling and dynamical analysis of spacecraft parachutes and parachute clusters*, Computational Mechanics, 48 (2011), pp. 345–364.

[33] S. Taneda, *Visual observations of the flow past a sphere at reynolds numbers between 104 and 106*, Journal of Fluid Mechanics, 85 (1978), pp. 187–192.

[34] T. Tezduyar, M. Behr, and J. Liou, *A new strategy for finite element computations involving moving boundaries and interfacesthe deforming-spatial-domain/space-time procedure: I. the concept and the preliminary numerical tests*, Computer Methods in Applied Mechanics and Engineering, 94 (1992), pp. 339 – 351.

[35] T. Tezduyar, M. Behr, S. Mittal, and J. Liou, *A new strategy for finite element computations involving moving boundaries and interfacesthe deforming-spatial-domain/space-time procedure: Ii. computation of free-surface flows, two-liquid flows, and flows with drifting cylinders*, Computer Methods in Applied Mechanics and Engineering, 94 (1992), pp. 353 – 371.

[36] T. E. Tezduyar, S. Sathe, R. Keedy, and K. Stein, *Space-time finite element techniques for computation of fluid-structure interactions*, Computer Methods in Applied Mechanics and Engineering, 195 (2006), pp. 2002–2027.

[37] T. E. Tezduyar, S. Sathe, M. Schwaab, J. Pausewang, J. Christopher, and J. Crabtree, *Fluid-structure interaction modeling of ringsail parachutes*, Computational Mechanics, 43 (2008), pp. 133–142.

[38] T. E. Tezduyar, K. Takizawa, C. Moorman, S. Wright, and J. Christopher, *Space-time finite element computation of complex fluid-structure interactions*, International Journal for Numerical Methods in Fluids, 64 (2010), pp. 1201–1218.

[39] B. Tutt, *The application of a new material porosity algorithm for parachute analysis*, in 9th International LS-DYNA Users Conference, 2006.

[40] B. Tutt, S. Roland, R. D. Charles, and G. Noetscher, *Finite mass simulation techniques in LS-DYNA*, 21st AIAA Aerodynamic Decelerator Systems Technology conference and Seminar, (2011).

[41] B. A. Tutt and A. P. Taylor, *The use of LS-DYNA to simulate the inflation of a parachute canopy*, 18st AIAA Aerodynamic Decelerator Systems Technology conference and Seminar, (2005).

[42] J. Wang, N. Aquelet, B. Tutt, I. Do, H. Chen, and M. Souli, *Porous euler-lagrange coupling: Application to parachute dynamics*, in 9th International LS-DYNA Users Conference, 2006.

[43] D. C. Wilcox et al., *Turbulence modeling for CFD*, vol. 2, DCW industries La Canada, CA, 1998.

[44] V. Yakhot and L. M. Smith, *The renormalization group, the epsilon-expansion and derivation of turbulence models*, Journal of Scientific Computing, 7 (1992), pp. 35–61.